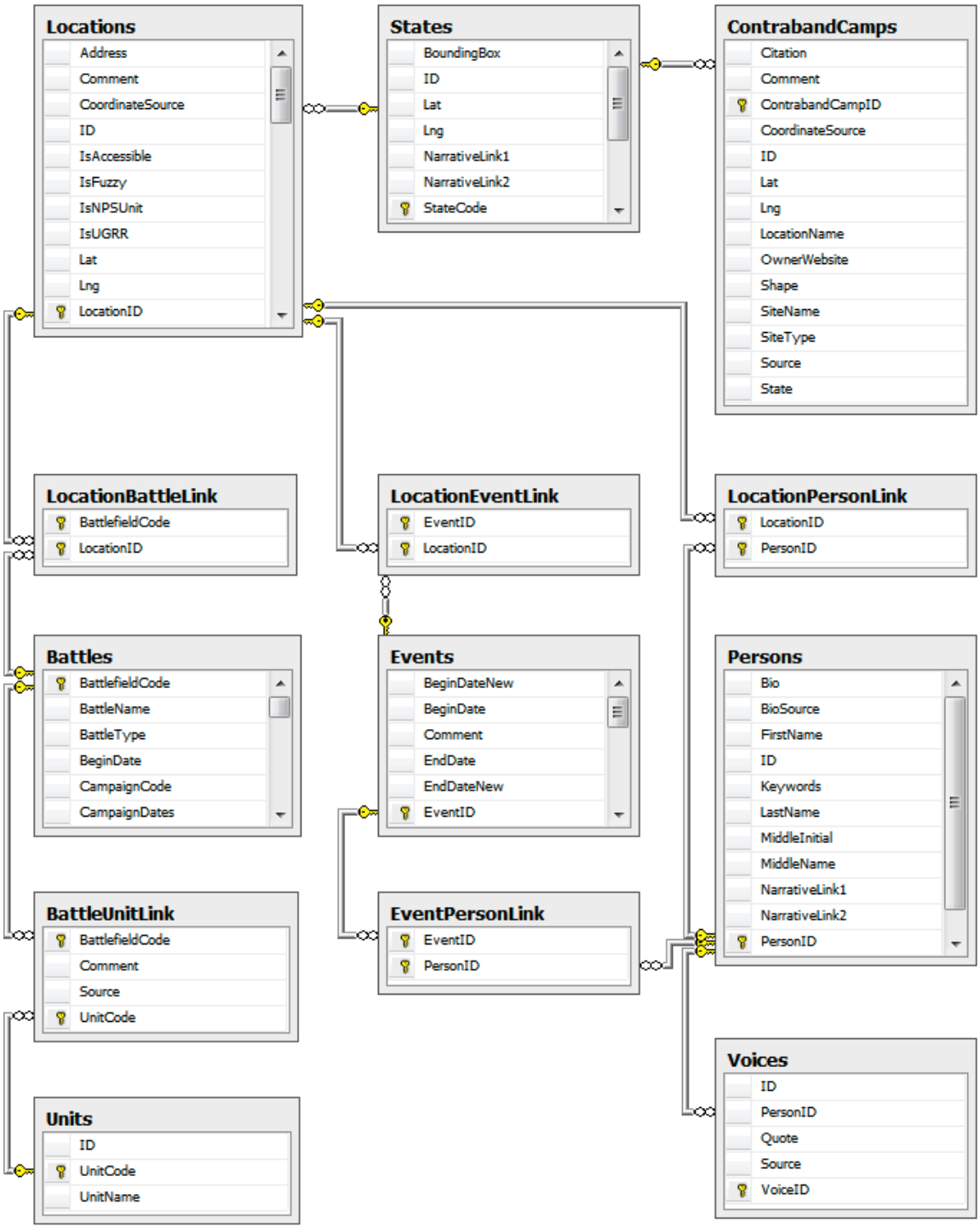
Using the Civil War 150th REST Service

First of all, here is a diagram of the database. This does not show all of the fields, but it does show the relationships between the entities. The REST service allows you to navigate these relationships. The many-to-many link tables that do have no non-key fields do not show up as entities in the service. They do, however, still act as navigation relationships in the service.



Debugging

There are a couple of tools you can use to browse the REST service. You can use a web browser, but if you want to return JSON data, you'll need to manipulate the request headers, and will want to use some sort of web debugging proxy. Fiddler is a good one for Windows. Charles is a good one that runs on both Windows and OS X.

JSON and XML

To request JSON data, you can either add "Accept: application/json" to the request header before sending the request or add a "$format=json" parameter to the URL. If you don't do one of these two things, you will get AtomPub-formatted XML back.

JSONP

The service also supports JSONP. This allows for cross-domain JSON requests. To let the service know that you want to it to format the response with a JSONP callback, you must specify the name of the callback in a "callback" parameter.

For example, if I wanted to request all of the "States" information back from the service and have it wrapped in a JSONP callback, here is the URL:

http://maps.nps.gov/civilwar150th/service.svc/States&$callback=myCallbackFunction

Here is a code snippet that utilizes jQuery and the JSON callback parameter to access the service:

```
$.getJSON('http://maps.nps.gov/services/civilwar150th/service.svc/States('VA'
)/?$format=json&$callback=?', function(data) {
      console.log(data);
});
```

Examples

Here are some example requests:

1. To access all of the data from the "States" table:

   http://maps.nps.gov/services/civilwar150th/service.svc/States

2. To access all of the data for the state of Virginia from the "States" table:

   http://maps.nps.gov/services/civilwar150th/service.svc/States('VA')

3. To access all of the "Locations" that are associated with the state of Virginia:

   http://maps.nps.gov/services/civilwar150th/service.svc/States('VA')/Locations

4. If you want get information about the state of Virginia and all of its “Locations” information back in a single request, you’ll want to use the “expand” query string option like this:

   http://maps.nps.gov/services/civilwar150th/service.svc/States('VA')?$expand=Locations

   This will return the related “Locations” entities in-line with the response of the parent “State”.

5. You can also use the “expand” query string option to traverse another relationship in a single request. This will get you back the data for the state of Virginia, all related “Locations” data, and all related “Battles” data in one single call:

   http://maps.nps.gov/services/civilwar150th/service.svc/States('VA')?$expand=Locations/Battles

   The return result is a nested hierarchy with all the data. If you are requesting JSON, the return will look something like this (I’ve taken out most of the data, but you can at least get the point from this):


```
{
    "d": {
        "__metadata": {
            "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/States('VA'
)",
            "type": "civilwarModel.State"
        },
        "ID": 25,
        "NarrativeLink1": null,
        "NarrativeLink2": null,
        "StateCode": "VA",
        "Summary": null,
        "SummarySource": null,
        "Title": null,
        "Lat": 37.9985033954151,
        "Lng": -79.4703096724197,
        "VirtualEarthZoom": 6,
        "ContrabandCamps": {
            "__deferred": {
                "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/States('VA'
)/ContrabandCamps"
            }
        },
        "Locations": [
            {
```

```
                "__metadata": {
                    "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'1db819ae-5c76-4052-ba62-000114380104')",
                    "type": "civilwarModel.Location"
                },
                "Comment": null,
                "CoordinateSource": "CRGIS",
                "ID": 3549,
                "IsAccessible": 1,
                "IsNPSUnit": 1,
                "Lat": 38.91228675,
                "Lng": -78.10826832,
                "LocationID": "1db819ae-5c76-4052-ba62-000114380104",
                "LocationName": "Manassas Gap",
                "LocationType": "Battlefield",
                "NationalRegisterID": null,
                "OwnerWebsite": null,
                "ShortSummary": null,
                "ShortSummarySource": null,
                "Summary": null,
                "SummarySource": null,
                "VirtualEarthZoom": null,
                "Battles": [
                    {
                        "__metadata": {
                            "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
108')",
                            "type": "civilwarModel.Battle"
                        },
                        "BattlefieldCode": "VA108",
                        "BattleName": "Manassas Gap",
                        "BattleType": "No Data",
                        "BeginDate": "7/23/1863",
                        "CampaignCode": "ME63-04",
                        "CampaignDates": "June-July 1863",
                        "CampaignName": "Gettysburg Campaign",
                        "Comment": null,
                        "EndDate": "7/23/1863",
                        "EnemyCasualties": null,
                        "EnemyName": "Confederate",
                        "EnemyTroopsEngaged": null,
```

```
                        "FirstEnemyCommander": "{306919F0-5AE8-4DEE-
BA37-0EA56B932B91}",
                        "FirstEnemyCommanderRank": null,
                        "FirstUSCommander": "{F7329508-919D-4FC2-AF29-
0DBB4C0F2291}",
                        "FirstUSCommanderRank": null,
                        "ID": 702,
                        "NarrativeLink1": null,
                        "NarrativeLink2": null,
                        "OwnerWebsite": null,
                        "Result": "Indecisive",
                        "SecondaryName": "Wapping Heights",
                        "SecondEnemyCommander": null,
                        "SecondEnemyCommanderRank": null,
                        "SecondUSCommander": null,
                        "SecondUSCommanderRank": null,
                        "ShortSummary": null,
                        "ShortSummarySource": null,
                        "State": "VA",
                        "Summary": "After recrossing the Potomac River
at Williamsport, Lee2019s army withdrew up the Shenandoah Valley.
Meade crossed the Potomac River east of the Blue Ridge and followed Lee
into Virginia. On July 23, Meade ordered the III Corps, under Maj. Gen.
William. H. French to cut off the retreating Confederate columns at
Front Royal by forcing passage through Manassas Gap.  At first light,
French began slowly pushing Walker2019s Confederate brigade
(Anderson2019s division) back into the gap. About 4:30 pm, a strong
Union attack drove Walker2019s men until they were reinforced by
Rodes2019s division and artillery. By dusk, the poorly coordinated
Union attacks were abandoned. During the night, Confederate forces
withdrew into the Luray Valley. On July 24, the Union army occupied
Front Royal, but Lee2019s army was safely beyond pursuit.",
                        "SummarySource": null,
                        "TheaterCode": "ME",
                        "TheaterName": "Main Eastern Theater",
                        "ThirdUSCommander": null,
                        "ThirdUSCommanderRank": null,
                        "TotalCasualties": 440,
                        "USCasualties": null,
                        "USTroopsEngaged": null,
                        "ThirdEnemyCommander": null,
                        "ThirdEnemyCommanderRank": null,
                        "Location": {
```

```
                            "__deferred": {
                                "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
108')/Location"
                            }
                        },
                        "BattleUnitLinks": {
                            "__deferred": {
                                "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
108')/BattleUnitLinks"
                            }
                        }
                    }
                ],
                "Events": {
                    "__deferred": {
                        "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'1db819ae-5c76-4052-ba62-000114380104')/Events"
                    }
                },
                "State": {
                    "__deferred": {
                        "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'1db819ae-5c76-4052-ba62-000114380104')/State"
                    }
                }
            },
            {
                "__metadata": {
                    "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'8f0923a6-6766-4c71-bab8-0006aec088ec')",
                    "type": "civilwarModel.Location"
                },
                "Comment": null,
                "CoordinateSource": "CRGIS",
                "ID": 4339,
                "IsAccessible": 1,
                "IsNPSUnit": 1,
                "Lat": 39.15001754,
```

```
                "Lng": -78.20509357,
                "LocationID": "8f0923a6-6766-4c71-bab8-0006aec088ec",
                "LocationName": "Kernstown I",
                "LocationType": "Battlefield",
                "NationalRegisterID": null,
                "OwnerWebsite": null,
                "ShortSummary": null,
                "ShortSummarySource": null,
                "Summary": null,
                "SummarySource": null,
                "VirtualEarthZoom": null,
                "Battles": [
                    {
                        "__metadata": {
                            "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
101')",
                            "type": "civilwarModel.Battle"
                        },
                        "BattlefieldCode": "VA101",
                        "BattleName": "Kernstown I",
                        "BattleType": "No Data",
                        "BeginDate": "3/23/1862",
                        "CampaignCode": "ME62-03",
                        "CampaignDates": "March-June 1862",
                        "CampaignName": "Jackson's Valley Campaign",
                        "Comment": null,
                        "EndDate": "3/23/1862",
                        "EnemyCasualties": 718,
                        "EnemyName": "Confederate",
                        "EnemyTroopsEngaged": 3800,
                        "FirstEnemyCommander": "{21D06A25-9C4F-4DEA-
A766-F60D20632999}",
                        "FirstEnemyCommanderRank": null,
                        "FirstUSCommander": "{CC0F3D5E-05E5-431E-82DC-
F7EF2A53CDB5}",
                        "FirstUSCommanderRank": null,
                        "ID": 695,
                        "NarrativeLink1": null,
                        "NarrativeLink2": null,
                        "OwnerWebsite": null,
                        "Result": "Union Victory",
                        "SecondaryName": null,
```

```
                        "SecondEnemyCommander": null,
                        "SecondEnemyCommanderRank": null,
                        "SecondUSCommander": null,
                        "SecondUSCommanderRank": null,
                        "ShortSummary": null,
                        "ShortSummarySource": null,
                        "State": "VA",
                        "Summary": "Relying on faulty intelligence that
reported the Union garrison at Winchester numbered only about 3,000,
201cStonewall201d  Jackson marched aggressively north with his 3,400-
man division.  The 8,500 Federals, commanded by Col. Nathan Kimball,
stopped Jackson at Kernstown and then counterattacked turning
Jackson2019s left flank and forcing him to retreat. Despite this Union
victory, President Lincoln was disturbed by Jackson2019s threat to
Washington and redirected substantial reinforcements to the Valley,
depriving McClellan2019s army of these troops. McClellan claimed that
the additional troops would have enabled him to take Richmond during
his Peninsula campaign.",
                        "SummarySource": null,
                        "TheaterCode": "ME",
                        "TheaterName": "Main Eastern Theater",
                        "ThirdUSCommander": null,
                        "ThirdUSCommanderRank": null,
                        "TotalCasualties": 1308,
                        "USCasualties": 590,
                        "USTroopsEngaged": 8500,
                        "ThirdEnemyCommander": null,
                        "ThirdEnemyCommanderRank": null,
                        "Location": {
                            "__deferred": {
                                "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
101')/Location"
                            }
                        },
                        "BattleUnitLinks": {
                            "__deferred": {
                                "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Battles('VA
101')/BattleUnitLinks"
                            }
                        }
                    }
```

```
                ],
                "Events": {
                    "__deferred": {
                        "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'8f0923a6-6766-4c71-bab8-0006aec088ec')/Events"
                    }
                },
                "State": {
                    "__deferred": {
                        "uri":
"http://maps.nps.doi.net/services/civilwar150th/service.svc/Locations(g
uid'8f0923a6-6766-4c71-bab8-0006aec088ec')/State"
                    }
                }
            }
        ]
    }
}
```


6. You can also do something similar to get to all of the “Events” that are associated to a particular state:

   http://maps.nps.gov/services/civilwar150th/service.svc/States('VA')?$expand=Locations/Events

7. Now, you (of course) don’t have to start at the “States” table. You can just as easily start at the “Locations” entity, or the “Battles” or “Events” entity. If you want to access an individual location with a unique identifier of “1db819ae-5c76-4052-ba62-000114380104”, you can use:

   http://maps.nps.gov/services/civilwar150th/service.svc/Locations(guid'1db819ae-5c76-4052-ba62-000114380104')

8. From there you can use the same syntax used above to query the different relationships. Here’s an example that traverses the relationship the opposite way, from “Locations” to “States.” Notice that “State” in the URL is not plural. This is because we are traversing a many-to-one relationship, and there can only be one “State” record for the “Location”:

   http://maps.nps.gov/services/civilwar150th/service.svc/Locations(guid'1db819ae-5c76-4052-ba62-000114380104')/State

9. You can use the “$select” query option to limit the data returned from the service. This is helpful if you only need a subset of information and want to limit the wire size of the data returned from the service.

   The following will return only the “StateCode” and “StateName” fields from the “States” entity:

http://maps.nps.gov/services/civilwar150th/service.svc/States?$select=StateCode,StateName

10. The "$select" query option also works on related entities. You must first, however, specify the name of the entity with a "/" to let the service know which entity the desired fields belongs to:

    http://maps.nps.gov/services/civilwar150th/service.svc/States?$filter=(StateName eq 'Virginia')&$expand=Locations&$select=Locations/Lat,Locations/Lng,Locations/LocationName

11. You can also use the "$count" query option to return the number of rows returned by a query:

    http://maps.nps.gov/services/civilwar150th/service.svc/States/$count

    It is important to note that the "$count" query option returns plain text, so if you set the request header as "application/json", the service will return an error.

Here is a list of all the query string options you can utilize to manipulate the data returned from the REST service, taken from the MSDN "Using Microsoft ADO.NET Data Services" document:

Option	Description	Example(s)
expand	The 'expand' option allows you to embed one or more sets or related entities in the results. This allows you to return related entities in-line with the response of the parent in a single HTTP request. You may specify multiple navigation properties to expand by separating them with commas, and you may traverse more than one relationship by using a dot to jump to the next navigation property.	--a customer with related sales orders /Customers('ALFKI')?$expand=Orders --a customer with related sales orders and employee information related to those orders /Customers('ALFKI')?$expand=Orders/Employees --Orders with related employees information and related shipper information /Orders(10248)?$expand=Employees,Shippers
orderby	Sort the results by the criteria given in this value. Multiple properties can be indicated by separating them with a comma. The sort order can be controlled by using the "asc" (default) and "desc" modifiers.	/Customers?$orderby=City /Customers?$orderby=City desc /Customers?$orderby=City desc,CompanyName asc
skip	Skip the number of rows given in this parameter when returning results. This is useful in combination with "top" to implement paging (e.g. if using 10-entity pages, saying $skip=30&top=$10 would return the fourth page). NOTE: Skip only makes sense on sorted sets; if an orderby option is included, 'skip' will skip entities in the order given by that	--return all customers except the first 10 /Customers?$skip=10 --return the 4th page, in 10-row pages /Customers?$skip=30&$top=10

	option. If no orderby option is given, 'skip' will sort the entities by primary key and then perform the skip operation.	
top	Restrict the maximum number of entities to be returned. This option is useful both by itself and in combination with skip, where it can be used to implement paging as discussed in the description of 'skip'.	--top 5 sales orders /Customers?$top=5 --top 5 sales orders with the highest TotalDue /Orders?$orderby=TotalDue&$top=5
filter	Restrict the entities returned from a query by applying the expression specified in this operator to the entity set identified by the last segment of the URI path.	-- all customers in London /Customers?$filter=City eq 'London' -- Match all Customers with the value of the property 'fullname' equal to 'Wayne, John' /Customers?$filter='Wayne, John' eq insert(ContactName, length(lastname), ',')
select	Restrict the fields returned from the service.	/Customers?$select=ContactName
count	Return a count of the records returned by a query.	/Customers/$count
Inclinecount		

The REST service also supports operators:

Operator	Description	Example
Logical Operators		
eq	Equal	/Customers?$filter=City eq 'London'
ne	Not equal	/Customers?$filter=City ne 'London'
gt	Greater than	/Product?$filter=UnitPrice gt 20
ge	Greater than or equal	/Orders?$filter=Freight ge 800
lt	Less than	/Orders?$filter=Freight lt 1
le	Less than or equal	/Product?$filter=UnitPrice le 20
and	Logical and	/Product?$filter=UnitPrice lteq 20 and UnitPrice gt 10
or	Logical or	/Product?$filter=UnitPrice lteq 20 or UnitPrice gt 10
not	Logical negation	/Orders?$filter=not endswith(ShipPostalCode,'100')
Arithmetic Operators		
add	Addition	/Product?$filter=UnitPrice add 5 gt 10
sub	Subtraction	/Product?$filter=UnitPrice sub 5 gt 10
mul	Multiplication	/Orders?$filter=Freight mul 800 gt 2000
div	Division	/Orders?$filter=Freight div 10 eq 4
mod	Modulo	/Orders?$filter=Freight mod 10 eq 0

Grouping Operators		
( )	Precedence grouping	/Product?$filter=(UnitPrice sub 5) gt 10

There are also functions that can be used with the filter query string operator:

String Functions
bool substringof(string p0, string p1)
bool endswith(string p0, string p1)
bool startswith(string p0, string p1)
int length(string p0)
int indexOf(string arg)
string insert(string p0, int pos, string p1)
string remove(string p0, int pos)
string remove(string p0, int pos, int length)
string replace(string p0, string find, string replace)
string substring(string p0, int pos)
string substring(string p0, int pos, int length)
string tolower(string p0)
string toupper(string p0)
string trim(string p0)
string concat(string p0, string p1)
Date Functions
int day(DateTime p0)
int hour(DateTime p0)
int minute(DateTime p0)
int month(DateTime p0)
int second(DateTime p0)
int year(DateTime p0)
Math Functions
double round(double p0)
decimal round(decimal p0)
double floor(double p0)
decimal floor(decimal(p0)
double ceiling(double p0)
decimal ceiling(decimal p0)
Type Functions
bool IsOf(type p0)
bool IsOf(expression p0, type p1)
<p0> Cast(type p0)
<p1> Cast(expression p0, type p1)

And finally, each of the REST service's supported data types have a literal form defined. This literal form is used in URLs generated and accepted by the service to identify the data type of a literal. Here are the literal forms for each supported data type:

Literal Form	EDM Type	CLR Type
null	null	null
binary'[A-Fa-f0-9][A-Fa-f0-9]*' **X '[A-Fa-f0-9][A-Fa-f0-9]*'** NOTE: **X is case sensitive and binary isn't. Spaces are not allowed between binary and the quoted portion. Spaces are not allowed between X and the quoted portion. Odd pairs of hex digits are not allowed.**	Edm.Binary	byte[]
true | false	Edm.Boolean	Bool
[A-Fa-f0-9]+	Edm.Byte	Byte
datetime'yyyy-mm-ddThh:mm[:ss[.fffffff]]' **NOTE: Spaces are not allowed between datetime and quoted portion. datetime is case-insensitive. The format actually allows anything from the DateTimeKind.RoundtripKind, which may preserve timezone information.**	Edm.DateTime	DateTime
decimal'[0-9]'	Edm.Decimal	Decimal
'[0-9]+ ([.[0-9]+] | [E[+ | -][0-9]+]) **Allows 1E+10**	Edm.Double	Double
'[0-9]+.[0-9]+f'	Edm.Float	Float
guid'dddddddd-dddd-dddd-dddddddddddd' **where each d represents [A-Fa-f0-9]** NOTE: **guid is case insensitive and spaces are not allowed between guid and the quoted portion.**	Edm.Guid	Guid
'[0-9]+'	Edm.Int16	Int16
'[0-9]+'	Edm.Int32	Int32
'[0-9]+L'	Edm.Int64	Int64
'[0-9]+'	Edm.SByte	SByte
'[0-9]+F'	Edm.Single	Single
'char*' NOTE: **Delimiters are escaped by doubling them.**	Edm.String	String